

RAČUNARSKA GRAFIKA

drugi kolokvijum

- 1) [25] **OpenGL:** Napisati na jeziku C ili C++ deo programa za crtanje scene prikazane na slici 1 primenom grafičke biblioteke OpenGL. Scena se sastoji od tri kocke, jedinične dužine stranica. Jedna se crta popunjenih stranica. Svaka stranica se boji različitom nijansom sive, a naspramne stranice su iste boje. Za preostale dve kocke crtaju se samo konture njihovih stranica, crnom bojom. Scena se crta na beloj podlozi. Koristi se ortogonalna projekcija, a vidljiva zapremina je definisana u sledećem opsegu: $x \in [-1.5, 1.5]$, $y \in [-1.5, 1.5]$, $z \in [0, 6]$. Napisati posebnu funkciju koja vrši osnovnu inicijalizaciju OpenGL sistema potrebnu za crtanje scene. Smatrati da je otvaranje prozora za crtanje realizovano u glavnom programu koji nije potrebno pisati. **Napomena:** rezultujuća slika ne sme da zavisi od redosleda crtanja kocki.



slika 1

- 2) [25] Postaviti jednu matričnu jednačinu koja određuje projekciju sa perspektivom iz centra projekcije $P(0,0,3)$ na projekcionu ravan $z=5$, slike proizvoljne tačke (u desnom 3D koordinatnom sistemu) u ogledalu postavljenom na ravan $z=y+4$. U sve matrice elementarnih transformacija koje učestvuju u matričnoj jednačini uvrstiti konkretne vrednosti. Nije potrebno množiti matrice.
- 3) [50] Odgovoriti koncizno (jedna do dve rečenice) na sledeća pitanja:
- Kojim redosledom se množe matrice pri računanju složene transformacije ako se 3D tačka predstavlja kao vektor-vrsta, a kojim ako se tačka predstavlja kao vektor-kolona?
 - Kojim projekcijama pripadaju i šta je osnovna karakteristika iskošenih (*oblique*) projekcija? Kako se nazivaju i koje su karakteristike podvrsta iskošene projekcije?
 - Koji parametri su potrebni da bi se odredila matrica orijentacije prikaza (transformacija iz koordinatnog sistema realnog sveta u koordinatni sistem prikaza) u SPHIGS-u?
 - Kako se može promeniti oblik nekog poliedra koji se nalazi u SPHIGS strukturi promenom nekog njegovog temena (npr. da se teme $(1,1,1)$ jedinične kocke se premesti u tačku $(2,2,2)$)?
 - Čemu služi pik-identifikator u paketu SPHIGS i da li više primitiva jedne strukture može imati isti pik-identifikator?

Napomene:

- Kolokvijum traje 120 minuta.
- Nije dozvoljena upotreba literature niti programabilnih kalkulatora.
- Dozvoljena je upotreba OpenGL podsetnika.

Rešenja zadataka

drugi kolokvijum 2008

1) Rešenje

```
void crtajStranu(float *boja)
{
    glBegin(GL_QUADS);
    glColor3fv(boja);
    glVertex2f( -0.5f, -0.5f );
    glVertex2f( 0.5f, -0.5f );
    glVertex2f( 0.5f, 0.5f );
    glVertex2f( -0.5f, 0.5f );
    glEnd();
}
```

```
void crtajKocku(float *boja1, float *boja2, float *boja3)
{
    crtajStranu(boja1);
    glPushMatrix();
    glTranslatef(0, 0, -1);
    glRotatef(180, 0, 1, 0);
    crtajStranu(boja1);
    glPopMatrix();
    //-----
    glPushMatrix();
    glTranslatef(0.5f, 0, -0.5f);
    glRotatef(90, 0, 1, 0);
    crtajStranu(boja2);
    glPopMatrix();
    glPushMatrix();
    glTranslatef(-0.5f, 0, -0.5f);
    glRotatef(-90, 0, 1, 0);
    crtajStranu(boja2);
    glPopMatrix();
    //-----
    glPushMatrix();
    glTranslatef(0, 0.5f, -0.5f);
    glRotatef(-90, 1, 0, 0);
    crtajStranu(boja3);
    glPopMatrix();
    glPushMatrix();
    glTranslatef(0, -0.5f, -0.5f);
    glRotatef(90, 1, 0, 0);
    crtajStranu(boja3);
    glPopMatrix();
}
}
```

```
void crtajScenu(void)
{
    glClear(GL_COLOR_BUFFER_BIT |
           GL_DEPTH_BUFFER_BIT);

    static float tamno_siva[]= { 0.3f, 0.3f, 0.3f };
    static float siva[]= { 0.6f, 0.6f, 0.6f };
    static float svetlo_siva[]= { 0.9f, 0.9f, 0.9f };
    static float crna[]= { 0, 0, 0 };

    glPushMatrix();
    glTranslatef( 0, 0, -3 );
    glRotatef(45, 0, 1, 0);
    glRotatef(45, 0, 0, 1);
    glPolygonMode(GL_FRONT, GL_FILL);
    crtajKocku(tamno_siva, siva,
              svetlo_siva);

    glPolygonMode(GL_FRONT_AND_BACK,
                 GL_LINE);
    glPushMatrix();
    glTranslatef(1, 0, 0);
    crtajKocku(crna, crna, crna);
    glPopMatrix();

    glPushMatrix();
    glTranslatef(0, -1, 0);
    crtajKocku(crna, crna, crna);
    glPopMatrix();
    glPopMatrix();

    glFlush();
}

void init()
{
    glClearColor (1.0, 1.0, 1.0, 0.0);
    glEnable(GL_DEPTH_TEST);

    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(-1.5f, 1.5f,
           -1.5f, 1.5f, 0, 6);

    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
}
```

$$2) [X' \ Y' \ Z' \ 1] = [X \ Y \ Z \ 1] * M$$

$$M = \text{Translacija1} * \text{Rotacija1} * \text{Ogledanje}_{xy} * \text{Rotacija2} * \text{Translacija2} * \text{Projekcija} * \text{Translacija3}$$

$$\text{Translacija1} = T(0,0,4) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -4 & 1 \end{bmatrix}$$

$$\text{Translacija2} = T(0, 0, 1) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

$$\text{Rotacija1} = R_x\left(\frac{\pi}{4}\right) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 0 \\ 0 & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\text{Projekcija} = P_p(0,0,-2) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\text{Ogledanje}_{xz} = S(1, 1, -1) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\text{Translacija3} = T(0,0,-5) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 5 & 1 \end{bmatrix}$$

$$\text{Rotacija2} = R_x\left(-\frac{\pi}{4}\right) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 \\ 0 & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Priznaje se i sledeće rešenje:

$$M = \text{Translacija1} * \text{Rotacija1} * \text{Ogledanje}_{xy} * \text{Rotacija2} * \text{Translacija2} * \text{Rotacija}_y(\pi) * \text{Projekcija}(0,0,2) * \text{Rotacija}_y(\pi) * \text{Translacija3}$$

$$\text{Rotacija}_y(\pi) = R_y(\pi) = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\text{Projekcija} = P_p(0,0,2) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -\frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$